

REMARKS/ARGUMENTS

Applicants request a phone interview to discuss this response.

The Examiner found that claims 11, 12, 46, 47, 57, and 58 would be allowed if rewritten in independent form. (OA5, pg. 2) Applicants submit that these claims are patentable over the cited art in their current form because the base claims from which they depend are patentable over the cited art for the following reasons.

1. Claims 1, 3-10, 31, 32, 37-45, and 48-56 are Patentable Over the Cited Art

The Examiner rejected claims 1, 3-10, 31, 32, 37-45, and 48-56 as obvious (35 U.S.C. §103(a)) over Bonner (U.S. Patent Pub. No. 2002/0029211) in view of Choy (U.S. Patent No. 5,551,027). Applicants traverse.

Claims 1, 37, and 58 concern accessing data in a database table, and require: receiving a fetch request to fetch data from a base table that satisfies a query predicate, wherein rows of the base table are stored in table partitions and wherein there is one index partition for each determined table partition, wherein each index partition includes nodes, wherein each node in each index partition includes at least one key column value from a corresponding table row in the table partition associated with the index partition and a location identifier identifying the corresponding table row in the corresponding table partition; comparing a direction indicated in the fetch request and an ordering of the index partitions; setting a fetch direction based on a result of the comparison of the direction indicated in the fetch request and the ordering of the index partitions; scanning the index partitions in the fetch direction to determine a set of nodes from the index partitions whose key column value satisfies the query predicate; ordering the set of determined nodes from the index partitions; selecting one node from the ordered set based on a position of the node in the ordering; and returning data from the table row identified by the location identifier in the selected node in response to the fetch request.

The Examiner cited para. 89 of Bonner as teaching the claim requirement of comparing a direction indicated in the fetch request and an ordering of the index partitions. (OA5, pgs. 3-4, 8) Applicants traverse.

The cited para. 89 mentions logic implemented in the database program 4 to return data to an application program 2 in response to a FETCH command, such as the FETCH command shown in FIG. 8. If (at block 252) the FETCH is "insensitive", then the database program 6

would position (at block 254) the cursor to the position specified in the FETCH operation, e.g., PRIOR, FIRST, LAST, CURRENT, etc. and then return (at block 256) the row at the new cursor position in the result table 50. If the returned row in the result table 50 was previously fetched with a FETCH SENSITIVE, it would reflect any changes made to the base table 60 prior to such FETCH SENSITIVE operation. The FETCH sensitivity that can be specified depends on the declared cursor sensitivity. If the cursor is declared as insensitive, then the FETCH can only be insensitive. However, if the cursor is declared as sensitive, then the FETCH can be declared as either sensitive or insensitive.

Although the cited Bonner discusses the operations of a fetch sensitive in which fetched data reflects changes to the base table prior to the fetch sensitive operation, the Examiner has not shown where para. 89 teaches the claim requirement of comparing a direction indicated in the fetch request and an ordering of index partitions. The cited para. 89 discusses declaring a cursor as “insensitive” or “sensitive” which indicates whether the fetched data from the result table should reflect changes to the base table since the fetch operations. The Examiner has not shown where para. 89 teaches that the direction indicated in the fetch request, e.g., forward or backward, is compared to an ordering of index partitions.

The Examiner cited the above discussed para. 89 of Bonner as teaching the claim requirement of setting a fetch direction based on a result of the comparison of the direction indicated in the fetch request and the ordering of the index partitions and scanning the index partitions in the fetch direction. (OA5, pgs. 3-4) Applicants traverse.

The cited para. 89 discusses a fetch insensitive in which data from the result table at the cursor position is returned and fetch sensitive where the returned data reflects changes to the base table prior to the fetch operation. This discussion of fetch “sensitive” and “insensitive” does not teach the claim requirement that the fetch direction is set based on the result of comparing the direction indicated in the fetch request and the ordering of the index partitions. The Examiner has not shown where para. 89 and the discussion of “fetch sensitive” teaches that the fetch direction is set based on comparing the direction indicated in the fetch request and the ordering of the index partitions. Instead, para. 89 discusses a fetch “sensitive” versus “insensitive”, which determines whether the data returned for the accessed row in the result table reflects changes made to the base table.

The Examiner cited Choy with respect the claimed structure of the index partition as including nodes, but did not site Choy with respect to the above discussed deficiencies of Bonner with respect to the claim requirements. (OA5, pg. 4) Thus, even if one were to combine the cited Bonner and Choy, the proposed combination does not teach or suggest all the claim requirements due to the above discussed deficiencies of Bonner with respect to the claim requirements of comparing the fetch request direction and index ordering and setting a fetch direction based on a result of the comparison.

Accordingly, claim 1, 37, and 48 are patentable over the cited art because the cited combination of Bonner and Choy do not teach or suggest all the claim requirements.

Claims 3-10, 31, 32, 38-45, and 49-56 are patentable over the cited art because they depend from claims 1, 37, and 48, which is patentable over the cited art for the reasons discussed above. Moreover, the below discussed dependent claims provide additional ground of patentability over the cited art.

Claims 3, 38, and 49 depends from claim 1, 37, and 48, respectively, and further require that the fetch direction is set opposite the direction indicated in the fetch request if the direction indicated in the fetch request is opposite the ordering of the index partitions.

The Examiner cited para. 90 of Bonner as teaching the additional requirements of these claims. (OA5, pg. 5). Applicants traverse.

The cited para. 90 mentions that if the FETCH is SENSITIVE, then the data manager 16 repositions (at block 258) the cursor (current row pointer) to the row in the result table 50 according to the operation specified in the FETCH statement, e.g., next, prior, first, k rows forward or backward in a relative or absolute operation, etc. The remaining para. 90 then discusses how to process a fetch sensitive when flags for the accessed row of the result table may indicate that the accessed row has been deleted from the base table.

If (at block 260) the delete flag in the CTL column in the accessed row of the result table 50 indicates a "delete hole", i.e., that the row has been deleted in the base table 60, then the executive 6 returns (at block 262) an error code that the requested row was deleted. Alternatively, if a "delete hole" is detected, then the data manager 16 may return to the base table to determine whether the delete was undone by a restore or savepoint operation. Otherwise, if the requested row has not been deleted, then the runtime 14 provides (at block 264) the data manager 16 the RID for the current row, accessed from the RID column of the determined row in the result table 50. If (at block 266) the data manager 6 returns a code indicating that the row in the base table 60 identified by the presented RID has been deleted, then the delete flag in the CTL column of the determined

row in the result table 50 is set (at block 268) to "on" (the runtime program 14 would instruct the data manager 16 to update the delete flag in the result table 50). Otherwise, the runtime 14 would receive (at block 270) the row identified by the RID in the base table 60 from the data manager 16.

The Examiner has not explained how the discussion in the cited para. 90 of how to handle a fetch sensitive when flags may indicate that the accessed row in the result table has been deleted in the base table teaches or suggests the claim requirement of setting the fetch direction in which the index partitions are scanned opposite the direction indicated in the fetch request if the ordering of the index partitions is opposite the direction in the fetch request. The Examiner has not explained how the discussion of how to handle a delete hole or flag indicating whether data to return from the result table has been deleted from the base table teaches or suggests determining whether to modify the fetch direction in which the index partitions are scanned based on whether a current fetch direction is opposite an ordering of the index partitions.

Accordingly, claims 3, 38, and 49 provide additional grounds of patentability over the cited art because the additional requirements of these claims are not taught or suggested in the cited art.

Claims 4, 39, and 50 depend from claims 1, 37, and 48, respectively, and further require that setting the fetch direction comprises: setting the fetch direction to backward if the fetch direction is backward and the fetch direction is not opposite the ordering of the index partitions or if the fetch direction is forward and the fetch direction is opposite the ordering of the index partitions; and setting the fetch direction to forward if the fetch direction is backward and the fetch direction is opposite the ordering of the index partitions or if the fetch direction is forward and the fetch direction is not opposite the ordering of the index partitions.

The Examiner cited the above discussed para. 90 of Bonner as teaching the additional requirements of these claims. (OA5, pg. 5) Applicants traverse for the following reasons.

As discussed, the cited para. 90 discusses how to handle a fetch sensitive where flags indicate that the accessed row in the result table has been deleted in the base table. The Examiner has not shown where the cited para. 90 teaches or suggests setting the fetch direction in which the index is scanned based on the partition index ordering and the direction indicated in the fetch request. Instead, the cited para. 90 discusses how to perform a fetch sensitive when

delete flags may indicate that the accessed row of the result table has been deleted in the base table.

Further, the Examiner has not cited any part of para. 90 that teaches or suggests setting the fetch direction to backward if the fetch direction is backward and the fetch direction is not opposite the ordering of the index partitions or if the fetch direction is forward and the fetch direction is opposite the ordering of the index partitions and setting the fetch direction to forward if the fetch direction is backward and the fetch direction is opposite the ordering of the index partitions or if the fetch direction is forward and the fetch direction is not opposite the ordering of the index partitions. The Examiner has not shown where the cited para 90 teaches or suggests the specific claim requirements of setting the fetch direction based on the fetch direction and the ordering of the index partitions is nowhere taught or suggested in the cited art.

Accordingly, claims 4, 39, and 50 provide additional grounds of patentability over the cited art because the additional requirements of these claims are not taught or suggested in the cited art.

Claims 5, 40, and 51 depend from claims 1, 37, and 48, respectively, and further require that if the fetch request is a first fetch of the fetch request, then selecting one node starting from one of: a lowest key value from each index partition if the fetch direction is forward or highest key value from each index partition if the fetch direction is backward.

The Examiner cited para. 100 of Bonner as teaching the additional requirements of these claims. (OA5, pgs. 5-6) Applicants traverse.

The cited para. 100 discusses a technique for prefetching pages including the previous rows in the backward direction from the current position in the result table. The data manager 16 would utilize an algorithm to determine whether a program is sequentially fetching backward sequential rows in the result table, such as if a certain number of the last pages fetched were in backward sequential order. In such case, the data manager 16 would use the space map to determine pages in the same segment or in other segments that include prior sequential rows in the database table from the current position and instruct the buffer manager 18 to prefetch into memory such determined pages in anticipation that they will be accessed during a sequential fetch backward operation.

Although the cited para. 100 discusses how to determine pages in a current or other segment including prior sequential rows from the current position, there is no teaching or

suggestion of selecting a node from a lowest or highest key value from each index partition depending on the fetch direction. The Examiner has not shown where cited para. 100 teaches selecting key values from index partitions as claimed. Instead, the cited para. para. 100 discusses how to use a space map to determine pages in the same or other segments including prior sequential rows.

Accordingly, claims 5, 40, and 51 provide additional grounds of patentability over the cited art because the additional requirements of these claims are not taught or suggested in the cited art.

Conclusion

For all the above reasons, Applicant submits that the pending claims 1, 3-12, 31, 32, and 37-59 are patentable over the art of record. Should any additional fees be required, please charge Deposit Account No. 09-0460.

The attorney of record invites the Examiner to contact him at (310) 553-7977 if the Examiner believes such contact would advance the prosecution of the case.

Dated: July 16, 2009

By: /David Victor/

David W. Victor
Registration No. 39,867

Please direct all correspondences to:

David Victor
Konrad Raynes & Victor, LLP
315 South Beverly Drive, Ste. 210
Beverly Hills, CA 90212
Tel: 310-553-7977
Fax: 310-556-7984